

Flight planning: node-based trajectory prediction and turbulence avoidance

Jacob C. H. Cheung*
Met Office, Exeter, UK

ABSTRACT: To tackle the ever-growing demand for air travel, the aviation industry has expressed great interest in assessing the impact weather uncertainty has on flight planning. However, due to the opacity of commercial flight planning systems, there has been an absence of a suitable research platform which weather centres could use to test and convey advancements in numerical weather prediction (NWP) to benefit operational flight planning.

In this paper, a simple yet versatile trajectory prediction system that aims to bridge the gap between NWP centres and the industry is proposed. The proposed system is based on A*, a node-based pathfinding approach which is simple to implement and configured to suit any requirements. Unlike analytical solutions, research findings from the proposed system can be readily implemented in commercial flight planning systems. An example of how clear air turbulence could have been considered before take-off and avoided is presented.

KEY WORDS A*; trajectory prediction; flight planning; turbulence; Dijkstra; pathfinding

Received 2 December 2016; Revised 10 March 2017; Accepted 29 March 2017

1. Introduction

Node-based pathfinding algorithms are designed to find the cheapest route between two vertices through a network of connected nodes. In aviation, different variants are adopted and tailored for use by air navigation service providers/flight dispatchers in operational flight planning. Commercial airlines are often interested in finding a route with the least operating cost (e.g. fuel, time or overflight charges) without compromising on safety for any given flight.

Trajectory prediction (TP) for commercial flights is currently performed using deterministic numerical weather prediction (NWP) as input, with the assumption that the wind forecast is accurate. It is therefore impossible to assess the uncertainties due to weather, which could have significant impact on many levels of operations. For instance, if the jet stream turns out to be stronger than predicted, scheduled eastbound flights will arrive early while westbound flights will have to detour around the unexpected headwind, causing congestion at busy airports such as Heathrow Airport (EGLL).

The aviation industry has recently expressed interest in assessing the impact weather uncertainty has on flight planning. It is difficult, however, to study the impact of NWP uncertainty on flight planning without access to their TP systems, which are commercially sensitive and not available to meteorological (Met) service providers. This is also one of the reasons why the aviation industry still relies on deterministic Met data whilst ensemble NWP has been made operational by major NWP centres over the last 10–20 years.

Whilst analytical solutions to time-optimal trajectories (Sawyer, 1950; Lunnon and Marklow, 1992; Kim *et al.*, 2015) exist in atmospheric science, they are fundamentally different from those in operational flight planning. It could therefore be

difficult for the industry to implement new findings from such test platforms in their existing TP systems. In addition, analytical solutions generally lack flexibility. For instance, extending an analytical solution to multiple flight levels or optimizing against a different parameter (e.g. fuel burn instead of time) would require derivation of the solution from scratch.

Given the opacity of commercial flight planning systems, the purpose of this study was to take the initiative and propose a simple TP algorithm that resembles the basics of those used for operational flight planning, with the hope of narrowing the gap in TP between atmospheric research and the aviation industry in the long run. In fact, the proposed algorithm has already been used as a research platform to allow Met service providers to assess the impact weather uncertainty has on flight planning and to introduce ensemble NWP to commercial flight planning (Cheung *et al.*, 2015). The same algorithm was also used to generate time-optimal flight trajectories in a separate study on the quantification of spatial spread in track-based applications (Cheung, 2016). This study describes in detail how the TP system is formulated and other considerations such as choice of grids and hazard avoidance.

2. Method

2.1. Choice of pathfinding algorithm

Pathfinding is an active research area in computer science and originated from the work of Dijkstra (1959). Given a network of nodes each with a non-negative cost function:

$$f(N_n) = \sum_{i=1}^n g(N_i) \quad (1)$$

where N_i denotes the i^{th} node along the path under investigation and $g(N_i)$ is the cost of going to node N_i from its parent node N_{i-1} , Dijkstra's algorithm (Dijkstra hereafter) guarantees the cheapest path between any two nodes. The algorithm works by

* Correspondence: J. C. H. Cheung, Met Office, FitzRoy Road, Exeter, Devon EX1 3PB, UK. E-mail: jacob.cheung@metoffice.gov.uk
This article is published with the permission of the Controller of HMSO and the Queen's Printer for Scotland.

calculating the lowest possible $f(N_n)$ (cheapest cost to travel from the starting node N_{start} to N_n), caching its immediate parent node N_{n-1} , and updating $f(N_n)$ for its neighbours iteratively until the destination node N_{dest} is reached. The solution can subsequently be retrieved by using N_{n-1} stored previously in each node as a tracer, working backwards from N_{dest} . Dijkstra has no knowledge about the structure of the node network and explores not only nodes that lead towards but also those that lead away from the destination. As a result, Dijkstra often explores more nodes than necessary and is not practical in most pathfinding applications.

It is possible to improve the efficiency of Dijkstra by specifying a heuristic function $h(N_n)$ (Hart *et al.*, 1968), which estimates the cost from N_n to N_{dest} . Equation (1) then becomes:

$$f(N_n) = h(N_n) + \sum_{i=1}^n g(N_i) \quad (2)$$

The heuristic function allows longer paths to be eliminated progressively during the search, effectively shortlisting the nodes that need examining before an optimal solution can be found. This algorithm, known as A*, guarantees an optimal solution like Dijkstra if the heuristic is admissible, i.e. never overestimates the actual cost, such that:

$$h(N_n) \leq h^*(N_n) \quad \forall n \quad (3)$$

where $h^*(N_n)$ is the true cost from N_n to N_{dest} . When $h(N_n) = h^*(N_n)$, A* explores the minimum number of nodes required for the optimal route to be found. Note that, as $h(N_n)$ decreases, A* becomes less efficient and takes longer to find the optimum path. When $h(N_n) = 0$, A* reduces to Dijkstra.

It should be mentioned that over the last few decades the concept of node-based pathfinding has been extended and variants have been proposed for different applications. These range from any-angle movements (e.g. Ferguson and Stentz, 2007; Nash *et al.*, 2007, 2009), fast replanning capabilities (e.g. Koenig and Likhachev, 2002; Koenig *et al.*, 2004), moving target points (e.g. Sun *et al.*, 2008, 2010) to fast suboptimal solutions (e.g. Botea *et al.*, 2004; Likhachev *et al.*, 2005). For simplicity and the purpose of the study, the proposed TP system presented is based on that of classic A*, with minimal modifications to allow movement in an arbitrary number of directions on a fixed flight level. Although it is possible to extend the idea to allow movement in the vertical, it is beyond the scope of this study due to the increased computing cost. In fact, for the same reason, major flight planning companies take an iterative approach where horizontal and vertical optimizations are performed separately.

2.2. Nodes

In addition to human-related factors (e.g. air traffic control (ATC)), weather has a significant impact on flight planning. It is therefore natural to define the nodes that represent the possible points A* can explore for an optimal path at the grid points specified by the output of the NWP model.

In classic A*, each node N_n ‘sees’ only the neighbouring nodes it is connected to and updates $g(N_n)$ and tracing parent as it is explored. In other words, the nodes do not carry information about their positions in the network, making it challenging to retrieve the appropriate weather data at each node.

In this study, all the nodes are defined in exactly the same way as in classic A* but with the geographical co-ordinates each is representing embedded. Note that this does not affect how A* works as the nodes are still clueless about the network. The retrieval of the relevant space–time weather data is handled in

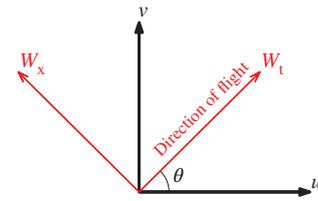


Figure 1. Transformation of zonal (u) and meridional (v) winds to tailwind–crosswind (W_t , W_x) components. [Colour figure can be viewed at wileyonlinelibrary.com].

the cost function (Section 2.3) just before the calculation of $g(N_n)$ and $h(N_n)$ as the latitude and longitude are now embedded. This approach benefits from the fact that, when extra weather fields are to be introduced in the TP model and hence the cost function, the rest of the algorithm stays unaffected.

2.3. Cost function

Pathfinding is essentially a minimization problem. The parameters of interest and the complexity of the cost function depend solely on the application and user’s requirements. For the purpose of this study, the cost function $g(N_n)$ is simply defined as the time to travel from N_{n-1} to N_n along the great circle with distance d at ground speed V_g :

$$g(N_n) = \frac{d}{V_g} \quad (4)$$

where

$$V_g = \sqrt{V_a^2 - W_x^2} + W_t \quad (5)$$

$$V_a = MS_0 \sqrt{\frac{T}{T_0}} \quad (6)$$

W_t and W_x represent the tailwind and crosswind respectively, T is the temperature, V_a is the true airspeed and S_0 is the speed of sound at T_0 . M is the Mach number. Subscript zero denotes that the specified variable takes its value at sea level. Assuming an aircraft travelling at an angle θ from the zonal direction as shown in Figure 1, W_t and W_x are given by:

$$\begin{bmatrix} W_t \\ W_x \end{bmatrix} = \begin{bmatrix} \cos \theta & \sin \theta \\ -\sin \theta & \cos \theta \end{bmatrix} \cdot \begin{bmatrix} u \\ v \end{bmatrix} \quad (7)$$

where u and v are the zonal and meridional wind components from the NWP model.

The definition of $g(N_n)$ in Equation (4) implies that any solution found will be time-optimal with respect to winds and temperature. For a balance between trajectory optimality and computing time, u , v and T are taken from the nearest time-step available from the NWP model output each time the cost function is invoked in the pathfinding phase.

2.4. Heuristic function

The heuristic function $h(N_n)$ estimates the cost of going from N_n to N_{dest} by assuming a direct great circle route and using Equations (4)–(6). Since an optimal solution is only guaranteed if $h(N_n)$ is admissible, a constant tailwind with zero crosswind is assumed in the calculation of $h(N_n)$. In other words, given a chosen value of W_t , the aircraft will always fly at the highest possible V_g , which in return underestimates the cost to finish the journey in a preferable way.

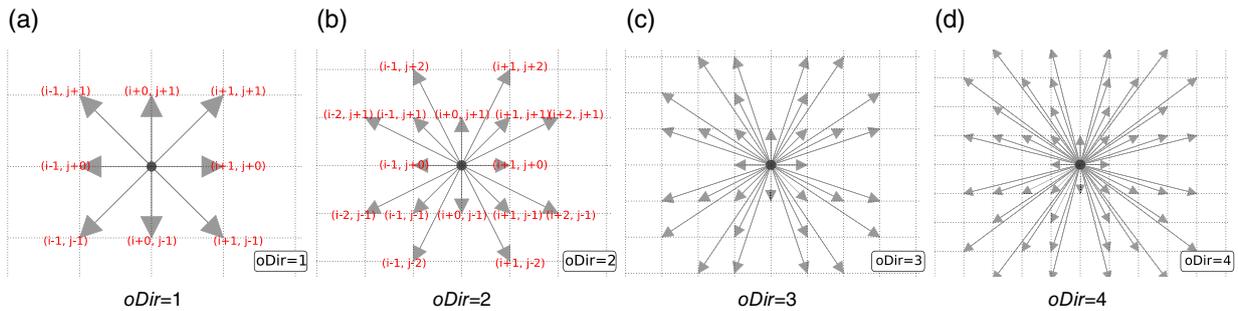


Figure 2. Illustration of neighbouring scheme with different values of $oDir$. Let $N(i, j)$ be the node located at the i^{th} longitude and j^{th} latitude of the gridded numerical weather prediction data (circle) and the arrowheads denote the neighbour nodes that $N(i, j)$ is connected to. The co-ordinates relative to the $N(i, j)$ are marked in (a), (b) but are left out in parts with higher $oDir$ (c), (d) for legibility. [Colour figure can be viewed at wileyonlinelibrary.com].

The choice of W_t depends on typical maximum wind speeds on a given flight level and area of interest. A high value of W_t would provide extra assurance of the optimality of the solution at the price of increased computing time. On the other hand, the TP model might fail to report the true optimum route if W_t is too low such that Equation (3) does not hold. A value of $W_t = 80 \text{ m s}^{-1}$ was verified to give true optimal solutions consistently at 250 hPa (not shown) and was used to calculate $h(N_n)$ in this study.

2.5. Neighbouring scheme

As described in Section 2.2, the nodes of the proposed TP system are assigned at grid points defined by the NWP data. It is therefore natural to register nodes immediately adjacent to a given node as its neighbours as shown in Figure 2(a) (hereinafter $oDir = 1$). This is analogous to many other pathfinding applications where the nodes are assigned at the grid points of a Cartesian grid.

Figure 3(a) shows the ‘optimum’ trajectory (solid) predicted by A^* for an aircraft going from Narita International Airport (RJAA) to EGLL under zero wind condition, which in theory should be identical to the great circle (dashed). In fact, it is a common issue that node-based pathfinding algorithms only offer suboptimal solutions in low-density node networks as the directions the algorithm can manoeuvre are heavily hindered by the neighbouring scheme. Such effect is especially prominent in the case shown, where an optimal route is to be found on a curved surface (the Earth) rather than a standard 2D Cartesian grid.

Although any-angle pathfinding algorithms do exist (e.g. Ferguson and Stentz, 2007; Nash *et al.*, 2007, 2009), which allow smoother solutions that resemble the true optimal route better, they are not always straightforward to implement. This is especially true in the case of this study, in which time-varying weather data have to be imposed.

In applications such as flight planning, where short computing time is favourable but not critical, it is possible to achieve smooth solutions using basic A^* alone, with minimal adjustment to the neighbouring scheme, which allows A^* to explore in 16 rather than 8 directions as in Figure 2(a). Given a node $N(i, j)$ where i and j denote its coordinates, all nodes within two grid points away that form an eigenvector with $N(i, j)$ are considered as neighbours. Rejecting nodes that share any existing eigenvectors prevents the algorithm exploring the same direction repeatedly and hence improves code efficiency. For example, the ‘east’ direction is represented by both $N(i+1, j)$ and $N(i+2, j)$ and therefore the latter is excluded. Examples

of neighbouring schemes with higher $oDir$ are also provided in Figures 2(c) and (d).

With $oDir > 1$, it is advised to resample the NWP data along each eigenvector under inspection and integrate the cost function to the associated neighbour node (see Figure 4). This will give a much better assessment of the cost of travelling from $N(i, j)$ to its neighbours that are not immediately adjacent (i.e. $N(i \pm m, j \pm n)$ where $m \geq 2$ and $n \geq 2$), especially in cases where the resolution of NWP data is coarse or when weather is highly variable in space and time. Figures 3(b)–(d) are the same as Figure 3(a) but for $oDir = 2, 3, 4$ respectively (cf. Figures 2(b)–(d)). It is obvious that there is an immediate improvement on the solution (solid) found by A^* , which resembles the great circle (true optimal solution under zero wind condition, dashed) much better. As $oDir$ increases, A^* has more freedom to explore in different directions and hence the solution resembles the true optimal route better. However, as there are more neighbour nodes to be explored at each iteration, the computing time is also longer. In the cases shown in Figure 3, the computing time has gone from 9 to 76 s as $oDir$ is increased from 1 to 4 on a standard desktop computer (benchmark times based on an implementation of A^* in Python, using only one of the eight available cores on an Intel Xeon E5-1620 CPU). Despite the increase, the actual computing time for $oDir = 4$ is still very short and perfectly feasible in applications like flight planning.

The benchmark times are only meant to highlight the trade-off between quality and performance. There are various ways to reduce the computing time further. For instance, implementation in languages like C++ or Fortran rather than Python as in this study would have decreased the computing time significantly. On the other hand, as multi-core CPUs have become widely available nowadays, parallel programming allows multiple neighbour nodes to be explored simultaneously and hence greatly improves on the performance.

3. Grid resolution

The number of nodes in a network dictates the number of possible routes between any given pair of nodes in the network. Although the heuristic function in A^* allows early elimination of costly paths, the increase in the number of possible routes with respect to available nodes is complex and nonlinear such that the search for an optimal solution could take an unreasonable amount of time.

Figures 5(a) and (b) show examples of a low resolution and a high resolution grid. The crosses denote the placement of

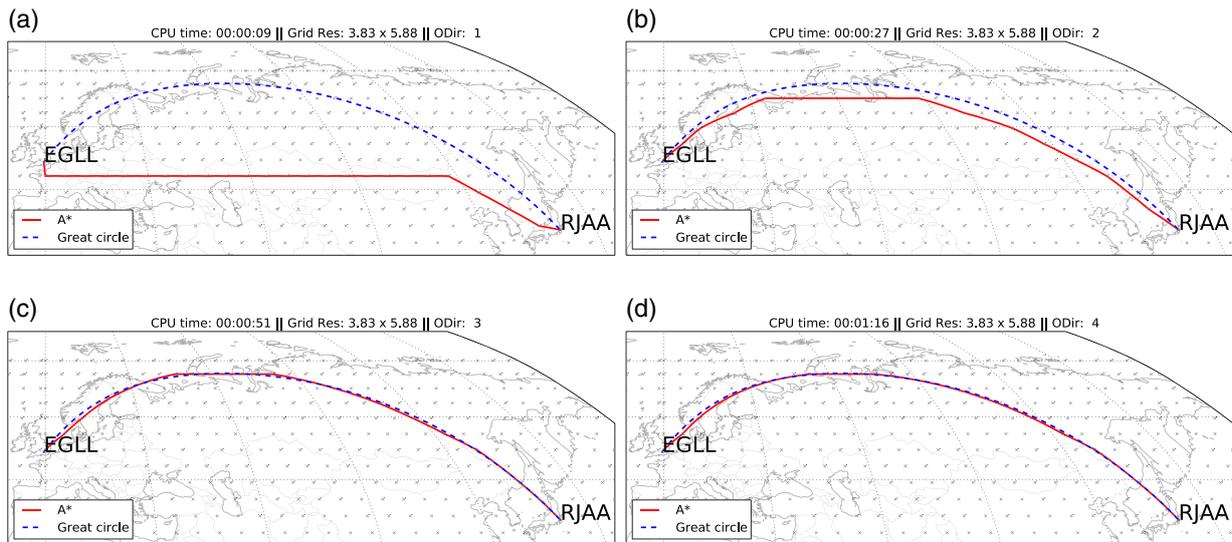


Figure 3. Great circle route (dashed) and trajectory predicted by the trajectory prediction model (solid) under zero wind condition from Narita International Airport (RJAA) to London Heathrow Airport (EGLL). (a)–(d) Each part shows the trajectory predicted using a different value of oDir , as depicted by the corresponding part in Figure 2. [Colour figure can be viewed at wileyonlinelibrary.com].

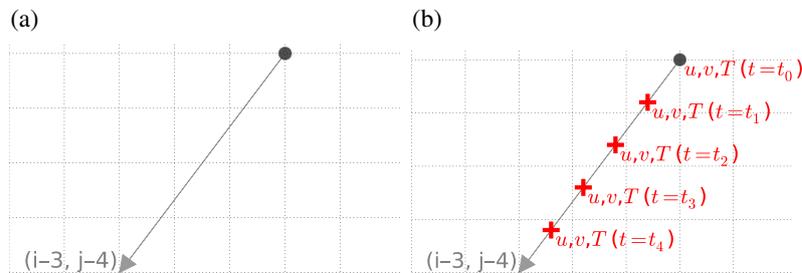


Figure 4. Graphical representation of how the cost function should be evaluated under the proposed neighbouring scheme with $\text{oDir} > 1$. Both parts show the eigenvector from a given node (circle) to one of its non-adjacent neighbours $N(i-3, j-4)$ (arrow). The example shown is for $\text{oDir} = 4$. In (a) the cost function is evaluated by noting the values of u, v, T at the origin node (circle) and assuming they stay constant along the eigenvector. In (b) u, v, T are first interpolated onto the sampling points (cross). The cost function is then evaluated progressively at each sampling point, using interpolated values of u, v, T at the correct space and time t until the target neighbour node is reached. [Colour figure can be viewed at wileyonlinelibrary.com].

the A^* nodes. For reference the great circles between Keflavik International Airport (BIKF) and ELLL and between John F. Kennedy International Airport (KJFK) and ELLL are shown. A dense network of nodes as shown in Figure 5(b) provides high manoeuvrability for A^* which implies that any solution found will be identical to or, if not, very close to the true optimal route. Referring to Figure 5(a), while the number of nodes between KJFK and ELLL would give A^* enough manoeuvrability, there is only a limited selection of nodes between BIKF and ELLL where A^* could explore. This implies that, in the latter case, any solution found is likely to be suboptimal compared to the true optimal route, despite being optimal in the given grid.

On the other hand, while a network with high node density as shown in Figure 5(b) would allow the true optimal route to be found for both airport pairs shown, it would take more computing power in the case of KJFK–EGLL as there are more nodes than necessary between the two airports. In other words, Figure 5(a) would be an appropriate choice for KJFK–EGLL and Figure 5(b) would be an appropriate choice for BIKF–EGLL.

Considering the fact that the grid resolution has a direct impact on the quality of the solution and performance, it is recommended to regrid the NWP data to an appropriate resolution before any

A^* node is assigned. A simple but effective regriding scheme, which dynamically determines a useful grid resolution for any given airport pair and available computing power, is proposed.

For a given airport pair, the grid resolutions r_{lat} and r_{lon} are determined by:

$$r_{\text{lat}} = \frac{\Delta_{\text{lat}}}{n_g} \quad r_{\text{lon}} = \frac{270}{(180/r_{\text{lat}}) - 1} \quad \text{if } |\Delta_{\text{lat}}| > |\Delta_{\text{lon}}| \quad (8)$$

$$r_{\text{lon}} = \frac{\Delta_{\text{lon}}}{n_g} \quad r_{\text{lat}} = \frac{180}{(270/r_{\text{lon}}) + 1} \quad \text{if } |\Delta_{\text{lat}}| \leq |\Delta_{\text{lon}}| \quad (9)$$

where Δ_{lat} and Δ_{lon} are the differences in latitude and longitude between the two airports respectively. n_g is the maximum number of grid points in any given direction. Figure 6 shows the route optimality (in terms of flight duration) and corresponding computing time as a function of n_g for ELLL–KJFK and ELLL–BIKF as an example. It is observed that the flight duration (route optimality) decreases (increases) and the associated computing costs increase as n_g increases. In both cases, the flight durations reach their minima at $n_g = 25$ while computing time continues to rise. The actual choice of n_g depends on the complexity of the weather (see the example in Section 5) and

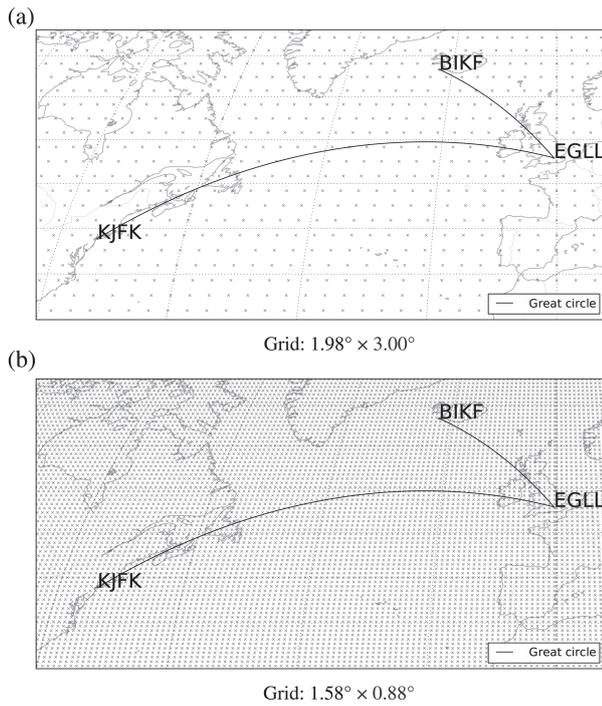


Figure 5. Examples of numerical weather prediction (NWP) grids at different resolutions. The crosses denote the grid points specified by the NWP which also mark the A* node locations. The solid lines show the great circles between John F. Kennedy International Airport (KJFK) and London Heathrow Airport (EGLL) and between Keflavik International Airport (BIKF) and EGLL. For reference, the World Area Forecast Centre grid resolution is currently $1.25^\circ \times 1.25^\circ$.

computing power available. For a good balance between optimality and performance, a value ≥ 25 is recommended for n_g in nominal weather conditions, based on extensive testing on various airport pairs (not shown).

4. Validation

For validation, TP hindcast was performed for all EGLL to KJFK flights in December 2016 to February 2017 from FlightAware using the baseline values of $\text{oDir} = 4$ and $n_g = 25$ described in previous sections. The wind and temperature fields used for TP were taken from the control member of Met Office Global and Regional Ensemble Prediction System (MOGREPS) (Bowler *et al.*, 2008). The MOGREPS data used are 3 h and have a horizontal resolution of 33 km. Only data from the 0000 Z run were used.

For each flight record, the TP hindcast with a validity time that was closest to the actual take-off time was used for validation. For instance, assuming a lead time of at least 24 h, a flight which took off at 1050 Z on 2 February 2017 would be compared against the $t + 36$ TP valid at 1200 Z on the same date. Similarly, a flight which took off at 1652 Z on 22 February 2017 would be verified against the $t + 42$ forecast valid at 1800 Z.

As the proposed TP does not account for delays due to ATC, flight records with a flight duration that lay outside two standard deviations and exhibited a holding pattern were discarded. It is inevitable, however, that some of the remaining flights still consist of a holding pattern but the impact on this study is minimized.

Figure 7 shows the flight times from EGLL to KJFK in December 2016 to February 2017. It is observed that the flight times

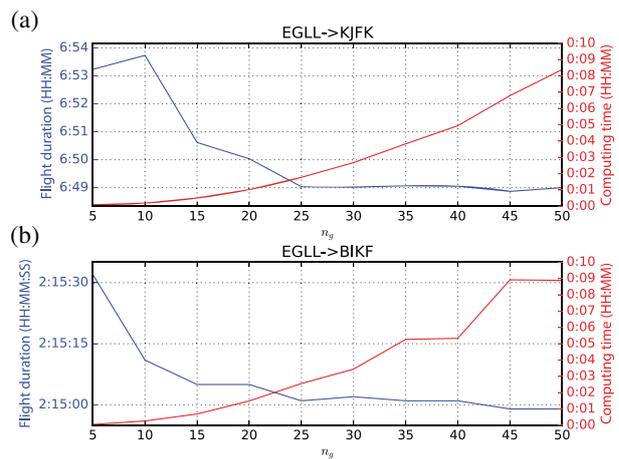


Figure 6. Route optimality in terms of flight durations (blue) and computing time (red) as a function of n_g . The example data shown are the trajectory prediction for a single flight from London Heathrow Airport (EGLL) to John F. Kennedy International Airport (KJFK) (a) and Keflavik International Airport (BIKF) and EGLL (b), taking off at 0000 Z on 28 February 2017.

predicted by the TP system (blue) have a high correlation value of $r = 0.911$ with the actual flight records (black). As the TP assumed a fixed cruise airspeed of 0.82 Mach for all phases of flight, the predicted flight times are generally underestimated. Assuming the errors are systematic, the TP output could be corrected by a least-squares fit (red):

$$t_{\text{actual}}(i) = \alpha t_{\text{predicted}}(i) + \beta \quad (10)$$

where $t_{\text{actual}}(i)$ and $t_{\text{predicted}}(i)$ are actual and predicted forecast flight durations for a given flight i . α and β are co-efficients to be determined by linear regression to minimize the root mean square error (RMSE) and were computed independently for each aircraft ID. For operation use, α and β could be continuously updated using previous flight records (e.g. from the last 3 months) for calibration purposes. It is important to note that the correlation co-efficient, which represents the performance of the TP, is unaffected. The RMSE after linear regression is 7.8 min.

5. Hazard avoidance/airspace blocking

In day-to-day operations, airspace might be closed completely for military training or events such as volcanic ash from the Eyjafjallajökull eruption in 2010 (Petersen, 2010). As these airspace constraints are static in space and time, they can be accounted for by setting $g(N_n)$ to infinity at the affected nodes.

On the other hand, flight hazards such as thunderstorms or turbulence could be taken into account in flight planning, when predicted in the NWP forecast. Although these are much more variable compared to the static airspace blocking requirements mentioned above, they could also be easily factored into the proposed TP system by changing $g(N_n)$, e.g.:

$$g(N_n) = \begin{cases} \infty & \text{if } \text{TI} \geq \text{TI}_{\text{threshold}} \\ d/V_g & \text{if } \text{TI} < \text{TI}_{\text{threshold}} \end{cases} \quad (11)$$

where TI is some form of turbulence indicator and $\text{TI}_{\text{threshold}}$ is the threshold value chosen by the user.

Figure 8 shows an example TP with $n_g = 50$ and $\text{oDir} = 4$ for a flight cruising at 0.82 Mach on 250 hPa from EGLL to Los

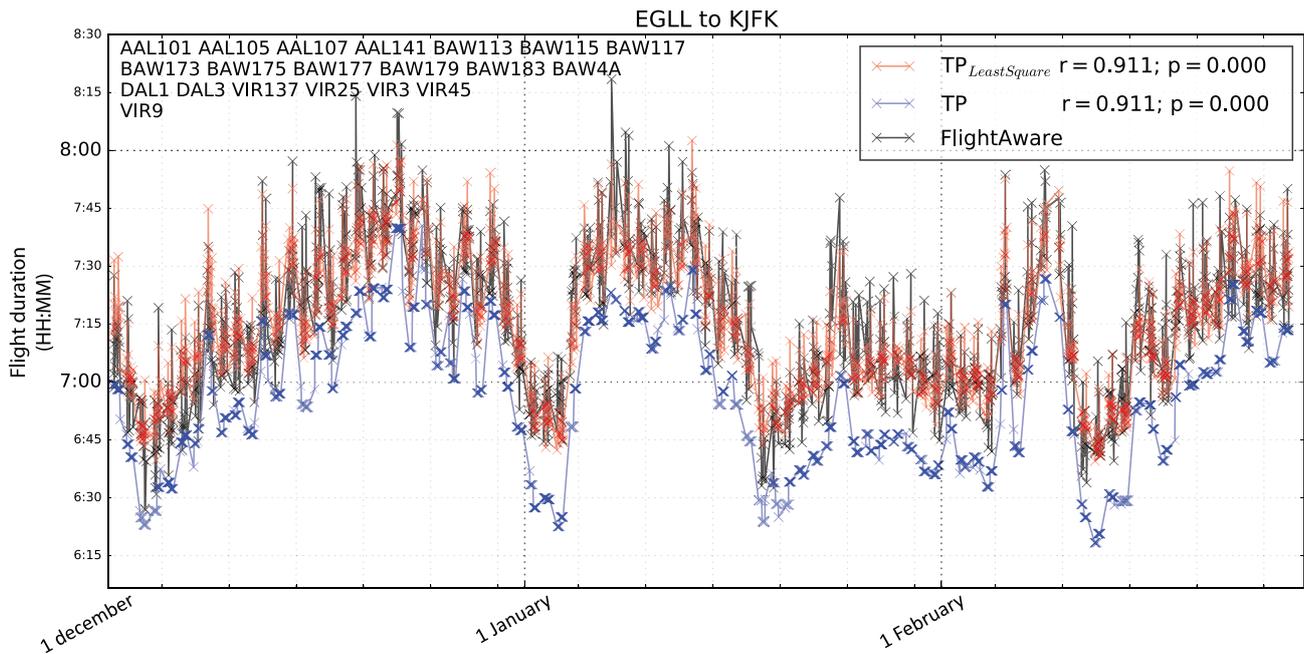


Figure 7. Time series of flight duration from London Heathrow Airport (EGLL) to John F. Kennedy International Airport (KJFK) in December 2016 to February 2017. The black line shows flight times reported by FlightAware. The blue line shows the flight times predicted by trajectory prediction (TP) with lead times of at least 24 h. The red line is the same as blue but is least-squares fitted to minimize root mean square error. The r and p values for the correlation between the predicted and the actual are shown. The IDs of the aircraft considered are listed in the top left.

Angeles International Airport (KLAX) with turbulence avoidance capability. The take-off date and time were chosen such that the unconstrained optimal path would coincide with the time and space of the turbulence encounter reported over Greenland in November 2014 (Elvidge *et al.*, 2017). The exact date and time of the encounter cannot be revealed for commercial reasons. The parts of Figure 8 show the Met Office World Area Forecast Centre (WAFc) clear air turbulence (CAT) (contour) and wind (quiver) forecast valid at 165, 210, 240 and 600 min after take-off. The WAFc CAT forecast is used as TI in Equation (11) in this example.

The green path shows the optimal trajectory in the unconstrained case, where turbulence is completely ignored in the pathfinding process. This is evident both in the southeast of Greenland in Figure 8(a) and over Greenland in Figure 8(b), where the green trajectory leads straight into regions with high CAT potential.

The yellow line shows the optimal trajectory with $TI_{\text{threshold}}$ set to 15. This implies that A^* will avoid regions at times when the turbulence potential exceeds the chosen threshold. It is observed that A^* takes a high-latitude trajectory which is about 4.5 min (0.74%) slower than the unconstrained case (see Figure 8 caption). It is worth mentioning that it is straightforward to apply the static blocking described above over northern Greenland to force A^* to find a route at lower latitudes, if the user prefers (not shown).

It is possible to reduce the risk of turbulence encounters further at the cost of flight duration/fuel, as shown by the trajectory in magenta where $TI_{\text{threshold}} = 10$. In this case, the path sets off in a similar way compared to the green line, but bends around instead of going straight through the turbulent region to the southeast of Greenland in Figures 8(a) and (b). After the first turbulence avoidance, A^* bends towards the south to avoid the region with high CAT potential predicted over Greenland (Figures 8(b) and (c)), which coincides with the turbulence reports received in

November 2014. It is expected that the time taken to fly along the magenta path is the slowest of all, taking 11.5 min (1.88%) more than the trajectory in green.

The trade-off between optimality of route and constraints applied, in this case risk of turbulence encounter, holds true as long as the heuristic is admissible. In other words, the unconstrained route (green) will never be more costly than any of the restricted (yellow and magenta) cases. The amount of trade-off, however, depends on the temporal and spatial span of the constraint and its proximity to the unconstrained solution.

In the case shown, one could argue whether or not it is worth making a horizontal detour, costing 1.88% more time and hence fuel burn, assuming constant airspeed and altitude throughout the flight. In fact, the actual extra fuel burn would be $> 1.88\%$ when climb and descent are taken into account.

Given that the horizontal extent of a turbulent volume is typically 100 times larger than its vertical extent (Sharman, 2016), a pilot will often slow down or request a climb/descent (subject to ATC approval) when CAT is encountered unexpectedly during flight. However, the decision making process is time critical and not straightforward. For climbing, the pilot would have to assess the coffin corner spread to make sure there is enough lift for the aircraft. For descent, the fuel burn would be higher at lower altitude due to denser air, plus extra fuel to bring the aircraft back to its optimum cruise altitude once past the turbulent region. Also, headwinds might be different at other flight levels which incur extra fuel burn. Taking the planned horizontal diversion, however, would take the stress off the pilot and ATC, with added certainty about extra fuel burn due to weather hazards.

Without engine specification data, it is difficult to assess the relative costs of taking a tactical diversion in the vertical *versus* planning a horizontal diversion before take-off, which also vary from case to case. Nevertheless, Figure 8 serves the purpose of demonstrating the hazard avoidance capability of the proposed system.

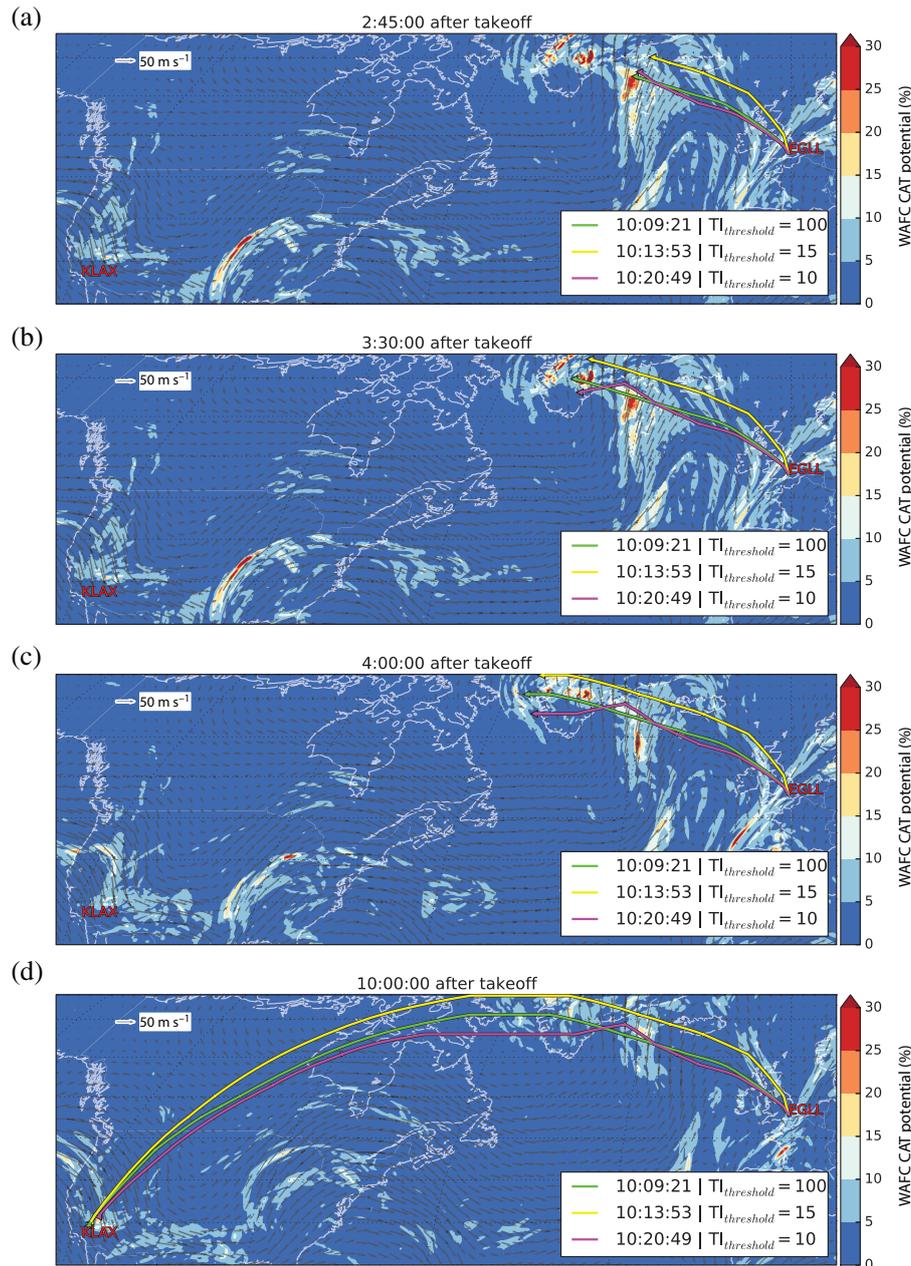


Figure 8. Examples of trajectory prediction (TP) for a given flight from London Heathrow Airport (EGLL) to Los Angeles International Airport (KLAX). (a–d) The parts show snapshots of the specified flight at different times after take-off. The take-off date and time are chosen such that the unconstrained predicted flight path (green) matches turbulence reports over Greenland on a given day in November 2014. The exact time and details are not revealed for commercial reasons. The quiver and contour plots show the wind and Met Office World Area Forecast Centre clear air turbulence (WAFC CAT) potential (%) forecast at least a day ahead of the event respectively. The trajectories shown are the time-optimal routes predicted by the TP system under different turbulence indicator thresholds $TI_{\text{threshold}}$. The total flight time for each route is shown.

Other than the example shown above, it is worth noting that any factor that might have an influence on the determination of the ‘best’ trajectory can be built into the cost function. For instance, airlines might choose to take a longer and less direct path to avoid high overflight fees. An easy way to achieve this would be to incorporate a penalty factor p in Equation (4) such that:

$$g(N_n) = p \frac{d}{V_g} \begin{cases} p > 1 & \text{penalized node} \\ p = 1 & \text{non-penalized node} \\ p < 1 & \text{preferred node} \end{cases} \quad (12)$$

6. Conclusion

The aim of this study was to propose a trajectory prediction (TP) system that resembles the basics of those used operationally for flight planning, with the long-term goal of closing down the gap in TP between atmospheric research and the industry in the future.

The proposed TP system is based on classic A*, a node-based pathfinding algorithm that is easy to implement and guarantees an optimal path along any grid provided that the heuristic function is admissible. Minimal modifications are proposed to render A* suitable for flight planning with gridded numerical weather

prediction (NWP) data, which help to maintain versatility and a low level of complexity of A^* . This is important as the definition of the 'best' route varies from user to user and is ultimately down to the user's own requirement. For example, some airlines might prefer to be punctual while others might seek to reduce costs wherever possible. The proposed TP system inherits the properties of A^* , meaning that any factors can be built into the cost function.

More importantly, the TP system described also provides a research platform for weather centres to test and translate any advancement in NWP (e.g. ensemble forecasts) into flight planning. Due to the resemblance to existing node-based commercial flight planning systems, research findings from the proposed system are readily usable in existing operational systems and hence much easier for the industry to adopt. In addition, rather than operating on a predefined set of waypoints, the proposed methodology promotes free routing, which coincides with SESAR's plan to move towards trajectory-based operations (<http://www.sesarju.eu>, accessed on 11 August 2016).

An example of TP for a time-optimal flight path from London Heathrow Airport to Los Angeles International Airport under different criteria for object avoidance was presented. The example was chosen to coincide with turbulence encounter reports over Greenland in November 2014 to demonstrate that the risk of turbulence encounter can be reduced by using the proposed TP system and space-time varying wind, temperature and World Area Forecast Centre clear air turbulence forecasts.

References

- Botea A, Müller M, Schaeffer J. 2004. Near optimal hierarchical path-finding. *J. Game Deve.* **1**: 7–28.
- Bowler NE, Arribas A, Mylne KR, Robertson KB, Beare SE. 2008. The MOGREPS short-range ensemble prediction system. *Q. J. R. Meteorol. Soc.* **134**: 703–722.
- Cheung JCH. 2016. Quantification of spatial spread in track-based applications. *Meteorol. Appl.* **23**: 314–319. <https://doi.org/10.1002/met.1556>.
- Cheung JCH, Brenguier J-L, Hally A, Heijstek J, Marsman A. 2015. Recommendations on trajectory selection in flight planning based on weather uncertainty. *Proceedings of the 5th SESAR Innovation Days*, Bologna.
- Dijkstra EW. 1959. A note on two problems in connexion with graphs. *Numer. Math.* **1**: 269–271.
- Elvidge AD, Wells H, Vosper SB, Cheung JCH, Derbyshire SV, Turp D. 2017. Moving towards a wave-resolved approach to forecasting mountain wave induced clear air turbulence. *Meteorol. Appl.* **24**: 540–550. <https://doi.org/10.1002/met.1656>.
- Ferguson D, Stentz A. 2007. Field D^* : an interpolation-based path planner and replanner. In *Robotics Research* Thrun S, Brooks R, Durrant-Whyte Hugh (eds). Springer Berlin Heidelberg; 239–253.
- Hart PE, Nilsson NJ, Raphael B. 1968. A formal basis for the heuristic determination of minimum cost paths. *IEEE Trans. Syst. Sci. Cybern.* **4**: 100–107.
- Kim J-H, Chan WN, Sridhar B, Sharman RD. 2015. Combined winds and turbulence prediction system for automated air-traffic management applications. *J. Appl. Meteorol. Climatol.* **54**: 766–784.
- Koenig S, Likhachev M. 2002. D^* Lite. *AAAI/IAAI*, 28 July–1 August 2002, Shaw Conference Center, Edmonton, Alberta, Canada; 476–483.
- Koenig S, Likhachev M, Furcy D. 2004. Lifelong planning A^* . *Artif. Intell.* **155**: 93–146.
- Likhachev M, Ferguson DI, Gordon GJ, Stentz A, Thrun S. 2005. Anytime dynamic A^* : an anytime, replanning algorithm. *ICAPS*, 5–10 June 2005, Monterey, CA; 262–271.
- Lunnon R, Marklow A. 1992. Optimization of time saving in navigation through an area of variable flow. *J. Navig.* **45**: 384–399.
- Nash A, Daniel K, Koenig S, Felner A. 2007. Theta * : any-angle path planning on grids. In *Proceedings of the National Conference on Artificial Intelligence*, Vol. 22. AAAI Press: Menlo Park, CA; MIT Press: Cambridge, MA; London, 1999; 1177.
- Nash A, Koenig S, Likhachev M. 2009. Incremental phi * : incremental any-angle path planning on grids. *Proceedings of IJCAI'09 Proceedings of the 21st International Joint Conference on Artificial Intelligence*, 11–17 July 2009, Pasadena, CA; 1824–1830.
- Petersen GN. 2010. A short meteorological overview of the Eyjafjallajökull eruption 14 April–23 May 2010. *Weather* **65**: 203–207. <https://doi.org/10.1002/wea.634>.
- Sawyer JS. 1950. Theoretical Aspects of Pressure-Pattern Flying. HM Stationery Office.
- Sharman R. 2016. Nature of aviation turbulence, in *Aviation Turbulence*. Springer; 3–30.
- Sun X, Koenig S, Yeoh W. 2008. Generalized adaptive A^* . *Proceedings of the 7th International Joint Conference on Autonomous Agents and Multiagent Systems-Volume 1*, International Foundation for Autonomous Agents and Multiagent Systems, 12–16 May 2008, Estoril, Portugal; 469–476.
- Sun X, Yeoh W, Koenig S. 2010. Moving target D^* Lite. *Proceedings of the 9th International Conference on Autonomous Agents and Multiagent Systems-Volume 1*, International Foundation for Autonomous Agents and Multiagent Systems, 9–14 May 2010, Toronto, ON, Canada; 67–74.